IEEE Software
www.computer.org/software

Threat Modeling: Diving into the Deep End

Jeffrey A. Ingalsbe, Louis Kunimatsu, and Tim Baeten, Ford Motor Co.
Nancy R. Mead, Software Engineering Institute

Vol. 25, No. 1
January/February 2008

Threat Modeling: Diving into the Deep End

Jeffrey A. Ingalsbe, Louis Kunimatsu, and Tim Baeten, *Ford Motor Co.*

Nancy R. Mead, *Software Engineering Institute*

Ford Motor Company is deploying Microsoft's Threat Analysis and Modeling methodology and tool to introduce threat modeling on strategically important IT applications and business processes.

Optimizing the working relationship between a company's IT security (ITS) group and its internal business customers is difficult at best. Who is responsible for security? What does "responsible" mean? For that matter, what does "security" mean? If ITS is solely responsible for security, as is often the case, then everything across the board will likely receive the same level of protection. In their defense, the members of ITS often don't know which asset means the most to the business, so the safest approach is to protect everything as much as possible.

This is sometimes called "peanut butter security." Such an approach has at least two problems: First, the members of ITS will never know when enough is enough; thinking the crown jewels are somewhere under the layer of peanut butter, they will want to make that layer as thick as possible. Second, ITS will always be resource constrained; its staff will always have to spend the maximum amount of time making the layer of peanut butter increasingly thicker, when they could be working on other things. Although it's certainly true that ITS should be solely responsible for some aspects of security, such as deploying antivirus software and updates, which should be invisible to its business customers, the latter should be full partners in securing the assets of the company.

Enter threat modeling, a process in which ITS and its business customers participate as full partners to better understand threats to assets and the vulnerabilities that make them evident. Using this process, ITS can understand which assets need the most or least protection, and apply the appropriate resources (staff, tools, and so on) accordingly.

Ford Motor Company is currently introducing threat modeling on strategically important IT applications and business processes. The objective is to support close collaboration between IT Security & Controls (the ITS group at Ford) and its business customers in analyzing threats and better understanding risk. To accomplish this, a core group of security personnel have piloted Microsoft's Threat Analysis and Modeling process and tool (http://msdn2.microsoft.com/en-us/security/aa570413.aspx) on a dozen projects. Here, we discuss this TAM process, its benefits and challenges, and some deployment solutions.

Threat modeling

Threat modeling is only one point on the broader risk management continuum. In one sense, it's a way of quantifying risk. A business assigns a risk rating to potential threats that the threat-modeling process has identified. The business can then prioritize actions on the basis of that assigned risk rating. Various groups in the security realm have given specific meanings to the generic term *threat modeling*. There are at least four different ways to sort threat-modeling methodologies:

- whether the methodology is systemic (focusing on the entire system) or not systemic (focusing on some subset of the entire system);

0740-7459/08/$25.00 © 2008 IEEE

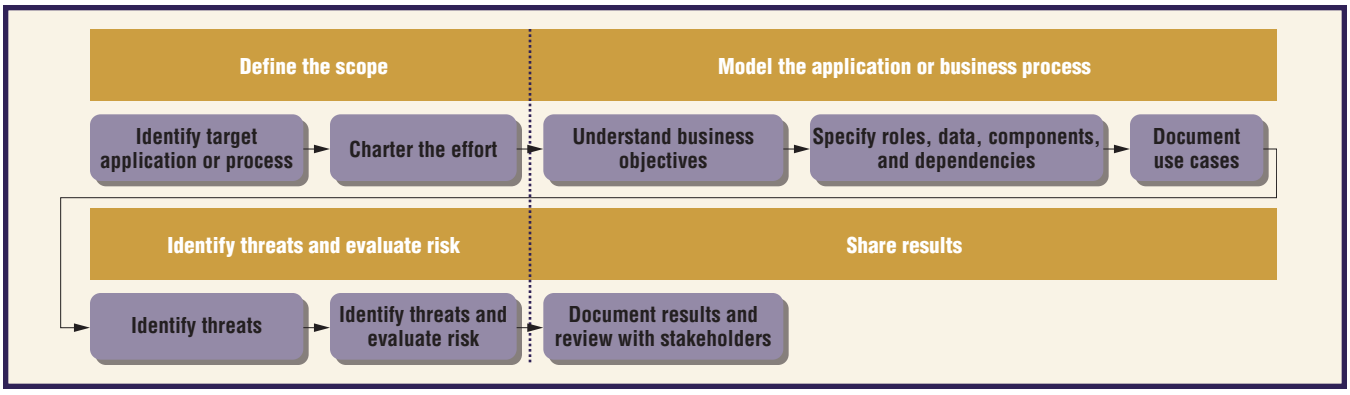

Figure 1. High-level threat-modeling process currently used at Ford Motor Company.

- whether it focuses only on valid-usage scenarios and labels invalid-usage scenarios as threats, or focuses on all possible usage scenarios and uses an adversary's goals, data flow, and trust boundaries to identify threats;
- whether or not the methodology is formal (mathematically); and
- whether or not the methodology focuses on automating the threat-modeling process.

Two threat-modeling methodologies originated at Microsoft. First, Frank Swiderski and Window Snyder described a systemic methodology (with an associated free tool) that focuses on data flow and trust boundaries.[1] Second, the Microsoft Application Consulting & Engineering (ACE) team developed its TAM methodology, which users can scope to a portion of the system and which focuses on a subset of the application's usage scenarios. The Trike methodology emphasizes an application's perspective and promotes automation.[2] Suvda Myagmar, Adam Lee, and William Yurcik discuss a methodology that (unlike the TAM process) is systemic and focuses on an attacker's perspective but (similar to the TAM process) heavily emphasizes the role threat modeling plays in the development of security requirements.[3] Common among all these methodologies is the emphasis on a systematic, repeatable process with products that enhance communication and highlight risk in a way that all stakeholders (even those without a security background) can understand.

The TAM process used at Ford

Ford's ongoing threat-modeling work is based on Microsoft's TAM process and tool. Discussions at Ford Motor Company about threat modeling for IT business applications and processes began in 2005. Faced with a growing demand for security services and constraints in resources and funding, the director of IT Security & Controls proposed threat modeling as a way to prioritize and focus efforts on specific projects with higher risk while postponing mitigation actions or accepting risk on projects with lower risk. Ford chose Microsoft's TAM methodology over Swiderski and Snyder's methodology because the former appeared more suitable for *line-of-business applications* (where intended usage is well defined), whereas the latter appeared more suitable for *products* (where intended usage is not well defined).

In 2006, two Ford IT Security & Controls engineers began working with a Microsoft consultant to evaluate and refine the use of the TAM process and tool. During that time, Ford conducted several pilot threat models. The engineers provided feedback to Microsoft about aspects of the tool such as usability and output formatting. Microsoft, in turn, incorporated some suggested improvements. Additionally, Ford reported defects to Microsoft, and some key fixes resulted. Ford is now using this improved tool and process in its ongoing threat-modeling efforts.

As figure 1 shows, this process consists of four high-level phases:

- define the scope,
- model the application or business process,
- identify threats and evaluate risks, and
- share results.

Define the scope

This phase selects the applications or business processes to undergo threat modeling and charters the necessary effort. The two major activities in this phase are as follows.

Identify target application or process. Thus far, IT Security & Controls has chosen targets for threat modeling in one of three ways: systems that scored high in a risk assessment survey; systems that executive-level managers in Ford's IT department deemed to be strategically important; and systems whose owners, architects, or managers thought threat modeling was a worthwhile thing to do. An organization could mandate threat modeling (a push strategy), or it could entice participation (a pull strategy).

Continuity in participation is paramount. It's essential that participants attend all sessions.

Although you could make an argument for using a push strategy, the advantages to growing threat-modeling advocates across the enterprise are far more significant.

Charter the effort. Assembling the team begins the chartering effort, but most of the work for this occurs during a kick-off meeting. This is the venue for helping participants understand what threat modeling is, what it isn't, why it's a good thing to do, what roles they'll play, how much time they'll spend, and what they must produce.

The first step is to define each participant's role in the process. (What we've discussed thus far might seem to imply that the target is an internal application or business process. But there's no reason why the target couldn't be a commercial-off-the-shelf product.) The roles and responsibilities identified by the threat-modeling team at Ford are as follows:

- *IT representative familiar with the application.* Constructing the model requires decomposing the application into its components, data, roles, and external dependencies. This IT representative is the person who manages the application on behalf of the business customer.
- *Business representative familiar with business objectives and the application's impact on the business.* Knowledge of business objectives and impact is necessary to evaluate risk and determine risk response.
- *IT Security & Controls representative.* A security and controls champion (SCC) with knowledge of and/or responsibility for the application fulfills this role. The SCC is an employee on the application development team who has background or expertise in IT security and controls. The SCC can help the development team by providing consultation regarding security-related issues and investigating potential security concerns associated with applications in the development team's portfolio. The SCC must be aware of threats resulting from the threat model. Participating in the process can also increase the SCC's awareness and knowledge of threat analysis and modeling, so that the SCC can facilitate future sessions and thus grow in competency.
- *Threat-modeling representative familiar with the tool, process, and facilitation.* This is an engineer familiar with the TAM tool and process. Currently, this is one of a few people in IT Security & Controls. The goal is to transition this expertise to the SCC.

The second step is to specify the necessary time commitment. Performing threat analysis and modeling requires four to five two-hour sessions. The number of sessions will correspond to the scope (narrow scope, fewer sessions; broad scope, more sessions) and the target's complexity. Typically, these working sessions are weekly, but two to three sessions per week is also an option. Although consolidating everything into one or two full-day sessions is possible, this approach could result in burnout and poorer quality output. In fact, in longer sessions (not even a full day), participants' attention began to wane after about two hours. Allowing time between working sessions enables reflection, informal discussion, and additional consideration outside the meeting's constraints.

Continuity in participation is paramount. It's essential that participants attend all sessions. Inconsistent attendance or the need to change representatives midstream causes disruption and redundancy, because it requires reviewing work from prior sessions to align participants. Also, remote participation isn't ideal. Optimizing attention and participation requires physical attendance at all working sessions. Although threat modeling can benefit distributed teams, participation seemed to decrease when Ford conducted sessions remotely.

The third step is to specify the deliverables that each participant must produce. All participants should contribute to the working sessions, but the threat-modeling representatives from IT Security & Controls produce the final report.

The fourth step is to define the scope of the threat model. It clarifies what participants did and did not look for during the threat-modeling exercise. At the kick-off meeting, the target application is already clear, but this step further narrows the scope by selecting a few use cases to analyze. Team members typically make this selection on the basis of the areas they're most concerned about (What keeps you up at night?) or the functional areas they use the most.

Ideally, the threat model scope would include all application (or business process) functionality (that is, use cases). However, owing to time and resource constraints, it's only feasible to include a subset of the use cases in the threat model. Choosing which use cases to include in the threat model is somewhat subjective, although every effort is made to choose those use cases having the potential to reveal the most risk.

Another consideration in use-case selection is the model's level of detail and the identification of redundant threats. Very similar use cases will most likely not result in identifying the same threats. Selecting a single use-case representative of other similar use cases can reduce this redundancy.

The fifth step is to prepare for the working ses-

sions. At the conclusion of the kick-off meeting, a participant from the team who will attend those sessions is asked to coordinate the schedule for them. This emphasizes the application team's ownership stake in the threat-modeling initiative. For convenience, the working sessions typically take place at the application team's location.

The working sessions serve to identify the application's primary business objectives, decompose the application into a predefined framework, document selected use cases, review identified threats, rank risk, and develop a risk response for each identified threat. These sessions comprise a series of structured interviews with the participants.

Model the application or process

This phase first identifies the business objectives that the application or business process fulfills. Next, it specifies the roles of people and services; the data created, read, or updated; significant components; and external dependencies for the application or business process. Finally, it documents use cases within the defined scope of the application or business process.

Understand business objectives. Part of the first working session focuses on understanding the business objectives that the application or business process meets. This isn't particularly time-consuming, but it is important. Later in the process, these business objectives will play a role in the evaluation and ranking of risk for each identified threat. A threat that has little or no impact on business objectives is less of a risk than one that can significantly impact the ability to achieve business objectives.

Specify roles, data, components, and dependencies. The balance of the first two working sessions is devoted to modeling the application. This model describes the application in terms of

- personnel roles,
- service roles,
- data,
- application components, and
- external dependencies.

The team documents descriptions and specific attributes associated with each of these entities. Using the TAM tool forces a level of consistency regarding which data is collected and the terminology used. It also provides a framework for the process, promoting consistency from one threat model to another.

After the team collects and documents the data, this data can serve to construct the data- and component-access matrices. These matrices facilitate the identification of access-control issues. These outputs improve the application's audit capability, providing a baseline that can be audited against.

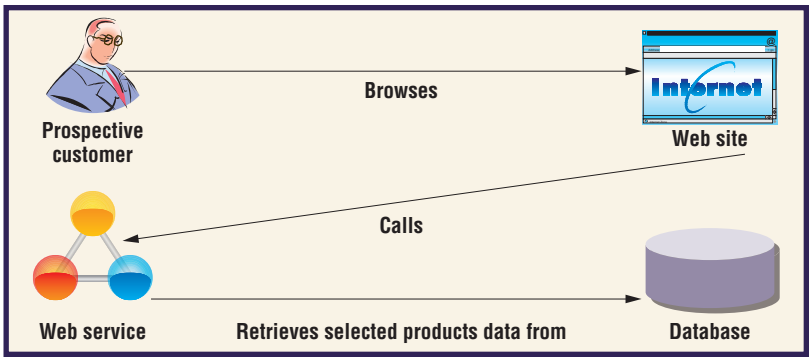

Figure 2. Use-case diagram. A role (prospective customer) performs some action (browsing) on the data (selected products), using specific components (Web site and database).

Document use cases. Typically, by the third working session, the team begins documenting the use cases selected during the kick-off meeting and enters them into the TAM tool. These documented use cases are sometimes called *usage scenarios* because they're less detailed than traditional use cases. The level of detail required to identify threats is less than that required to develop the application. Again, some variability and subjectivity might enter into the process of documenting use cases. Here, the threat modelers from IT Security & Controls rely on their experience to capture the appropriate level of detail to identify threats. This group documents the use cases as a series of tasks or steps involving interaction between entities contained in the model. A *role* is either a person (user) or system entity that performs or causes some action. (In figure 2, the role is a "prospective customer.") *Data* is a high-level, or conceptual, data element. (In figure 2, the data is "selected products.") A *component* is a logical functional unit within the process or system being modeled. (In figure 2, both the Web site and the database are components.)

Identify threats and evaluate risk

This phase involves identifying threats and analyzing them to rank risk.

Identify threats. When the steps of the first use case are documented, the TAM tool can generate a list of threats associated with that use case. The TAM tool identifies three threats for each step in the use case: one each for confidentiality, integrity, and availability.

Analyze threats and evaluate risk. The team reviews each threat and ranks them according to risk, evaluating the latter in terms of severity (impact on the business) and likelihood of occurrence. Here, the

After quantifying the risk, the team develops the risk response to the specific threat.

team employs the DREAD framework,[4] which uses the *d*iscoverability, *r*eproducibility, and *e*xploitability subfactors to determine the likelihood of a threat occurring, and uses the *a*ffected users and *d*amage potential to measure that threat's severity. The TAM tool gives participants three to five predefined responses to choose from for each subfactor (for example, for the damage potential, the possible responses are trivial, minor, moderate, major, and critical). Each possible response has a corresponding numeric value, which the TAM tool uses to derive the risk ranking associated with the threat.

Note that the threat is never implied to be malicious; it can be manifested inadvertently. The severity question might simply ask, "If an unauthorized disclosure occurs at this step, what would be the impact on the business and who would be affected?" or "If inaccurate data is provided at this step, what would be the impact on the business and who would be affected?" Separating likelihood from impact is often difficult. When asked to rate impact, participants often respond with "that would never happen"—especially when the application has been in production for a significant amount of time without the threat occurring. The facilitator must focus the discussion on each measurement distinctly with a question such as, "We know it might never happen, but what if it did?"

Participants representing the business provide the primary input for two severity and impact questions. Application and security team representatives do the same for the three likelihood questions. However, all participants are involved in the discussions. On the basis of their responses, the DREAD plug-in on the TAM tool derives a numeric risk value from 1 to 9 (1 being the lowest risk, and 9 the highest) for each threat.

After quantifying the risk, the team develops the risk response to the specific threat. Threats with low associated risk can be accepted. Moderate- to high-risk threats can be reduced through mitigation actions or avoided (with possible modification to the process), or options to transfer those risks might be sought.

The obvious benefit is the identification of previously unrecognized threats. But, even for previously considered threats, the mitigation action plans often don't correspond to the threat's risk level. The advantage of using the threat model is to apply (or not apply) the appropriate mitigation corresponding to the risk level.

The cycle of documenting steps, reviewing and ranking threats, and developing risk responses is repeated for each selected use case. The number of use cases selected for analysis determines the required number of working sessions. Typically, a two-hour work session is sufficient to analyze two to three use cases, depending on their complexity.

During use-case analysis, discussion might reveal a relationship or integration with another functionality not fully covered in the use case. The team might decide to expand the scope to include additional use cases. However, it's important to focus only on the target asset and not expand the scope so much that the team ends up analyzing external dependencies (other applications, infrastructure functionality) in depth.

Share results

After the analysis of all selected use cases and the completion of all the working sessions, IT Security & Controls prepares a final report consisting of an executive summary and detailed findings documenting the results. The group reviews these items in a one-hour meeting with the rest of the team. Management sponsors not directly participating in the working sessions can benefit from attending the final-report review because it summarizes the session work and reviews the risk response and justification for all threats ranked as medium or high risk.

The executive summary contains

- the objectives of the specific threat analysis and modeling;
- a brief description of the process employed;
- a brief description of the scope;
- any identified issues pertaining to data- or component-access control;
- issues identified during the working sessions that might not be apparent from reviewing the identified threats individually;
- a summary of all threats ranked as medium or high risk (organized by pertinence to confidentiality, integrity, and availability) containing a description of the specific threat, the risk ranking, the risk response (accept, reduce, transfer, or avoid), and the justification for the risk response; and
- a roster of participants and a session schedule.

The detailed-findings section contains

- a detailed description of what was modeled (business objectives, personnel roles, service roles, components, data, external dependencies);
- a description of the use cases analyzed; and
- a detailed list of all identified threats (associated use cases, risk ranking, factors used in risk ranking, risk response, description of justification of risk response, data- and component-ac-

cess control matrices, subject-object matrix, an appendix containing a glossary of terms, and operational definitions for factors used in risk quantification).

This final report identifies threats and associated risks ranked on a scale of 1 to 9. If a threat has a known attack mode and associated mitigation action, this document includes that information. Otherwise, this report doesn't include recommended mitigation actions. Further consultation with IT Security & Controls personnel is as an option but is not included in threat analysis and modeling.

Challenges and some successes

In many ways, the challenges to introducing threat modeling in a large, global organization are no different from those involved in any new IT initiative.

Organizational change

Evolving from a policy-based IT security posture to one that increases consideration of risk is difficult, owing to the variability inherent in a risk-based approach. Policy is typically easier to comprehend and tends to be binary (compliant versus noncompliant), which in turn facilitates governance. A risk-based approach is more variable, because it depends on specific circumstances. Risk-based analysis can be more time-consuming and can be susceptible to subjective, emotional discussion and decision making. The benefit is a "right sizing" of mitigation actions as opposed to a common requirement for all IT assets regardless of specific circumstances. This avoids expending time and resources on mitigations in which the associated risk might not merit it.

Funding

Challenges with funding vary between organizations but always involve some sort of rationalization of investment on the basis of perceived value. However, as with most investments in IT security, justification comes from cost avoidance, which is notoriously difficult to quantify. At the start of the initiative, finding a champion and starting with small proofs of concept overcame these challenges. Demonstrated successes and testimonials provided justification for continued investment. At some point, institutionalizing threat modeling might be desirable. More significant funding challenges will then present themselves because every project will require additional time and resources. One solution to this challenge would be to scale down the current threat-modeling methodology on the basis of some criterion. Another solution would be to develop a metrics program to bolster the justification; however, at Ford, that work has just begun.

Commitment

These challenges involve justifying a commitment of hours over a period of time, usually over and above current responsibilities. Threat-modeling exercises take about 10 hours of meeting time for each participant. Add to that any research work or investigations that arise from the meetings. On the surface, a commitment of 10 hours might seem small, but if the initiative is voluntary and not mandatory, asking for this commitment is more challenging. At the end of the threat-modeling exercise, most participants are advocates. Their understanding of the business impact of the threats to the assets they manage is enough to convince them. However, whether there's an upper threshold beyond which even an advocate of the process will balk isn't yet clear.

Often, during the course of the working sessions, there is an experience of group enlightenment referred to as the "aha moment." These moments occur during the documentation or review of use cases (and at other times) when the modeling exercise illuminates a misalignment or mixed understanding among the participants. Suddenly, they come to a common understanding of a process task or a new perspective on a potential business impact. When these moments occur, the participants find greater value in their participation and renewed interest in the working sessions.

Commitment challenges that involve the internal business customers can be even more problematic. However, the advantages of having their input on the value of assets and the business impact of threats realized is so significant that virtually any amount of work required to procure their involvement is justified.

Language

Currently, the vocabulary and taxonomy of threat modeling is IT biased. This impedes communication with, and understanding by, internal business customers. Telling the story in a concise, transparent manner is critical to accurately evaluating risk. Different reporting formats and terminology should be considered, evaluated, and tested to improve conveyance.

Just another security initiative

It would be nice if every security initiative was successful for and transparent to internal business customers, but that's simply not the case. The latter are typically wary of any conversation that begins

About the Authors

Jeffrey A. Ingalsbe is a senior IT Security & Controls engineer with Ford Motor Company. His technical interests include information security solutions for the enterprise, threat modeling, and strategic security research. He received his MS in computer information systems from the University of Detroit Mercy. Contact him at 17475 Federal Dr., Suite 800-D04, Allen Park, MI 48101; jingalsb@ford.com.

Louis Kunimatsu is a senior process methodologist and integrated Six Sigma Black Belt in IT Security & Controls at Ford Motor Company. His research interests include risk management, process design and metrics, and business modeling. He received his BA in communications and media arts from the University of Michigan. Contact him at Fairlane Business Park I, Suite 800, 17475 Federal Dr., Allen Park, MI 48101; lkunimat@ford.com.

Tim Baeten is a senior security architect in IT Security & Controls at Ford Motor Company. His research interests include international IT standards (especially in the area of Web services), threat modeling, and information assurance. He received his BS in management information systems from Oakland University. Contact him at Fairlane Business Park I, Suite 800-B41, 17475 Federal Dr., Allen Park, MI 48101; tbaeten@ford.com.

Nancy R. Mead is a senior member of the technical staff in the Survivable Systems Engineering Group of the Computer Emergency Response Team (CERT) Program at Carnegie Mellon University's Software Engineering Institute. She is also a faculty member in the Master of Software Engineering and Master of Information Systems Management programs at Carnegie Mellon University. Her research interests include information security, software requirements engineering, and software architectures. She received her PhD in mathematics from the Polytechnic Institute of New York. She is a fellow of the IEEE and the IEEE Computer Society, and a member of the ACM. Contact her at SEI, 4500 Fifth Ave., Pittsburgh, PA 15213; nrm@sei.cmu.edu.

with "Hi, I'm from ITS and I'm here to help." The only real solution is to establish an ongoing trust relationship, and this takes some effort.

The threat-modeling initiative is in its infancy. Several significant challenges remain, but it's already clear that the process has value. It has revealed previously unknown threats and business impacts, and it has encouraged risk-based discussions with internal business customers. Future work should focus on the following:

- considering threat modeling later in the life cycle,
- reducing variability in process execution, and
- integrating threat modeling in the software development life cycle.

Although the optimal time, in terms of cost-effectiveness, to conduct threat analysis and modeling is early in the life cycle, conducting this analysis later in the development life cycle still has significant value. Doing so increases security awareness for application development and business participants and promotes their understanding of the value of mitigations. Additionally, it provides line-of-sight between threats and impacts to the business objectives. Thus, post-design threat modeling provides several opportunities. First, it can help analyze a control deficiency identified in some other manner (controls reviews, audits, and so forth). An appropriately scoped threat model can provide a better understanding of the risk associated with the deficiency and help determine the extent of mitigation actions. Second, conducting post-design threat modeling on applications planned for outsourcing can provide more focused security requirements for vendor evaluation and selection. Third, adapting and applying the threat-modeling process can be useful for analyzing service and business processes.

Reducing variability in process execution is another important area. Because of limited experience, such variability has resulted in inconsistent data collection. More disciplined process execution should yield higher integrity data for analysis. Data resulting from threat analysis could identify infrastructure requirements or foundational mitigation actions that reduce risk of threats common to multiple applications. Such data could also identify or quantify varying risk levels between business organizations (lines of business). This risk might already be informally acknowledged by groups with an enterprise view, but analysis of data resulting from threat analysis could provide greater detail and insight. Threat analysis data could also quantifiably illuminate and emphasize which IT assets are of greater value or have more impact on business objectives. Finally, such data can better gauge an organization's appetite for risk.

Integrating threat modeling in the software development life cycle is another major area for future work. This integration would eliminate subjectivity and variability in the process of selecting targets and use cases, because all new development would execute the process. The resulting data would enable the types of analyses described earlier in this article.

References

1. F. Swiderski and W. Snyder, *Threat Modeling*, Microsoft Press, 2004.
2. P. Saitta, B. Larcom, and M. Eddington, "Trike v.1 Methodology Document [Draft]," 13 July 2005, www.octotrike.org/Trike_v1_Methodology_Document-draft.pdf.
3. S. Myagmar, A. Lee, and W. Yurcik, "Threat Modeling as a Basis for Security Requirements," *Proc. Symp. Requirements Engineering for Information Security* (SREIS 05), 2005, www.sreis.org/SREIS_05_Program/short30_myagmar.pdf.
4. M. Howard and D. LeBlanc, *Writing Secure Code*, 2nd ed., Microsoft Press, 2002.